

競合型共進化アルゴリズムにおけるフォーカシングに対する 適応度共有法の適用

佐藤 竜也 有田 隆也

名古屋大学 大学院情報科学研究科

抄録 競合型共進化アルゴリズムのフォーカシング問題とは、特定遺伝子にのみ進化圧がかかる現象であり、良解を得ることを妨げる。特に2集団の場合は、個体の評価が相手集団の個体の表現型に依存するため、一方の集団の個体の表現型多様性の偏りが相手集団個体の評価の偏りを生み、フォーカシングを発生させる。本研究では、このような問題を、表現型多様性の制御機能を一方の集団に持たせ、相手集団に対し適切な進化圧を与えて解決する。多様性制御手法として、遺伝子型/表現型の類似度ではなく、振る舞いの履歴の類似度を用いた適応度共有法を提案する。表現型多様性の維持により相手集団の安定した進化を実現することを従来手法との比較評価により確認した。

Application of Fitness Sharing to Focusing on Competitive Coevolutionary Algorithms

Tatsuya Sato and Takaya Arita

Graduate School of Information Science, Nagoya University

Abstract In the framework of competitive coevolutionary algorithms, focusing refers to the ability of coevolving opponents to challenge one another by testing weak dimensions of performance. In particular, since the fitness of individuals depends on the relationship with opponents in phenotype space, biased diversity of phenotype in one population also biases diversity of phenotype in another population, and it causes focusing in negative sense. This paper proposes a method of fitness sharing based on the similarity in behavior of individuals, which aims to solve this issue by creating proper evolutionary pressure for one population. Comparative evaluation using two tasks shows that it makes evolution stable by maintaining adequate phenotypic diversity.

1. はじめに

共進化とは、ある生物の種とその同種・他種間の相互作用によって現れる進化のことである。これに基づいた共進化アルゴリズムは、従来の進化的計算のように静的に与えられた適応度関数を用いるのではなく、個体同士が影響を及ぼしあうことによって適応度が決定する。ある個体の適応度はそれを評価するための集団によって違いため、適応度地形は世代に応じて動的に変化する。また、その相互作用の定義によって競合型、協調型の2種類に分けられる。協調型は個体間の関係が相利共生であり、競合型は一方の適応度の増加が他方の減少につながる。

競合型共進化アルゴリズムの代表的な適用例に、ホス

ト/パラサイト共進化型遺伝的アルゴリズム(HPGA)によるソーティングネットワークの設計⁴⁾があげられる。ホスト個体にソーティングネットワーク、パラサイト個体としてそのホスト個体を評価するためのテストデータをおき、両集団を共進化させることで効率の良いソーティングネットワークを得た。その他に、分類問題に対するニューラルネットの進化¹⁰⁾、セルオートマトン密度分類問題¹¹⁾⁸⁾⁷⁾やミニマックス問題への適用¹⁾⁵⁾、ゲームプレイヤーの戦略の進化¹³⁾¹²⁾などがある。これらは、従来の遺伝的アルゴリズムを適用した場合に比べ、良い解を得た。

競合型共進化アルゴリズムの特長として、以下の2点があげられる。

- 両集団が互いに適度な選択圧をかけるので、局所

解に陥りにくい。

- 上記ソーティングネットワークのように、解候補の完全な評価が困難な問題において、部分的な評価で探索可能。

しかし、共進化アルゴリズムによる探索を困難にする性質として、適応度が相手集団に依存した相対的なものであるため、推移律が成立しないという点がある。これは、個体間の優劣の關係にサイクルが存在しうることを表す。Watson らは、個体間の相互作用を単純な 2 人ゲームとして表現したナンバーズ・ゲームを提案し、その上で競合型共進化アルゴリズムに起きる問題点として以下の 3 つをあげた¹⁵⁾。

(1) 遊離問題

集団間の進化の歩調が大幅に離れ、集団内の個体間の適応度差がなくなり、選択がほぼランダムになるため、進化が進みにくくなること。

(2) 堂々巡り

進化が起きているように見えるが、推移律が成立しないため、同じような個体が出現する状況を繰り返すこと。

(3) フォーカシング問題

ある特徴を表す遺伝子にのみ進化圧がかかり、バランスの取れた良い解が得られないこと。

遊離問題の解決方法として、逆噴射機能を有する HPGA がある¹⁶⁾。これはパラサイト集団の適応度の平均が宿主集団の適応度の平均よりある一定以上超えたときに、評価の低いパラサイトほど適応度が高くなるように適応度を変換するものである。

堂々巡りに対する解決方法として、殿堂入り¹³⁾ や ナッシュ・メモリ²⁾ などのメモリ・メカニズムがある。これらの方法は、各世代で得られた最も適応度の高い個体をメモリに確保し、それを適応度評価に使うことで獲得した形質が失われるのを防ごうというものである。

フォーカシング問題の解決には、集団に多様性を持たせることが必要であると指摘されている⁶⁾¹¹⁾。その多様性維持手法として資源共有法が提案されている¹³⁾⁷⁾。これは相手個体による評価から得られる利得を、その相手から同じ評価を得た個体同士で分配することにより、多様性を得ようという方法である。また、適応度共有法による探索の効率化¹⁴⁾、多目的最適化の概念を用いたパレート共進化⁹⁾がある。

なお、推移律が成立しない問題において堂々巡りが起きる場合、その原因はフォーカシングによって偏る形質の変化に由来すると考えることができる。よってフォーカシング問題の解決が堂々巡りの解決につながる

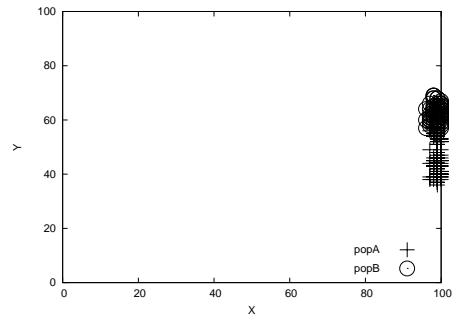


図 1 ナンバーズ・ゲームにおけるフォーカシングの例

る可能性があると考えられる。

本研究ではフォーカシング問題のより効率的な解決法として、遺伝子型/表現型空間での近さではなく、個体の振る舞いの履歴の類似度を使って多様性を保つ適応度共有法を提案する。従来手法と提案手法の評価実験を、ナンバーズ・ゲームとセルオートマトン (CA) 密度分類問題の上で行った。各手法の比較評価により、その有効性を示す。

2. フォーカシング問題

2.1 フォーカシングの例

図 1 に、2 次元のナンバーズ・ゲーム (4.1 で詳述) に基本的な競合型共進化アルゴリズムを適用した際にフォーカシングが発生した例における個体の分布図を示す。図中、popA と popB は各集団中の個体を表す。ナンバーズ・ゲームの純粋戦略は 2 次元平面上の各点であり、2 点間の差が最小である座標の値の大小を比較し、大きい方が利得を得る。ナンバーズ・ゲームにおいて理想的な進化は、x 座標も y 座標も大きい値になることだが、図 1 では、x 軸方向に偏った進化をしている。

図 2 は、各集団の x 座標、y 座標の平均値の推移をプロットしたものである。380 世代付近から両集団の x 座標の平均値の差が小さく、そして y 座標の差が大きくなっており、競合のほとんどが x 座標の比較により決定している。そのため、x 座標値の大きい個体ほど適応度が高くなり、それらが次世代に選択されやすくなる。同図から、それ以降では x 軸方向への過剰な進化が発生していることが分かる。x 軸方向へのフォーカシングが発生している間、y 座標に関する評価がされず、y 座標の変化は突然変異によるランダムウォークである。

フォーカシング問題の解決方法として、個体の評価すべき特徴の偏りを防ぐことがあげられる。共進化アルゴリズムでは、競合する集団中の表現型が互いに

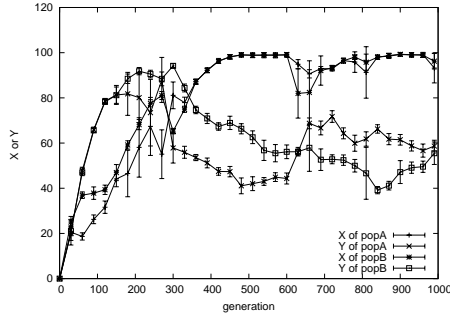


図 2 各集団の座標の平均値の変化

相手の適応度地形を決定しているため、一方の集団中の表現型多様性を維持することにより、フォーカシング問題を解決できると考えられる。

2.2 適応度共有法

フォーカシング問題を解決するための多様性を制御する方法として、一般的な遺伝的アルゴリズムなどで用いられる適応度共有法³⁾の導入が考えられる。適応度共有法とは、集団中の個体に多様性を持たせるために、遺伝子型/表現型空間上に分布する個体の密度に比例してその部分の適応度を下げる適応度スケールリングである。個体が適応度地形のある1つの解候補の周辺に密集した場合、適応度共有法は一時的にその周辺の適応度を下げ、他の解候補の適応度を相対的に上げる。この操作により、それらが次世代に残る確率が上がり、多様性が生まれる。

個体 i のスケールリング前の適応度を $F(i)$ 、スケールリング後を $F_{fs}(i)$ とした時、適応度共有法によるスケールリングを以下に定義する。

$$F_{fs}(i) = \frac{F(i)}{\sum_{j \in P} sh(d(i, j))} \quad (1)$$

ここで関数 sh はシェアリング関数と呼ばれるもので、空間上で個体間が与える影響の度合いを表す。この関数の出力は、個体 i と j が同一である場合には1、そうでない場合は距離 d に比例して0に近づく。シェアリング関数は、一般に以下のものが用いられる。

$$sh(d) = \begin{cases} 1 - (d/\sigma_{share})^\alpha & \text{if } d < \sigma_{share} \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

σ_{share} は個体間が影響を与え合う距離の閾値、 α は距離が及ぼす影響の度合いを調整するパラメータである。

一般に距離関数 $d(i, j)$ の定義により適応度共有法は、遺伝子型 (G-FS)、表現型 (P-FS) の2種類に分けられる。

2.3 資源共有法

共進化アルゴリズムにおける多様性を制御する方

法として資源共有法 (Resource Sharing, RS) がある¹³⁾⁷⁾。ある相手個体による評価から得られる利得は、従来の共進化アルゴリズムでは自集団中の他個体の影響を受けることはないが、RS ではその相手個体に対して同じ評価を得た個体同士で利得を分配する。ある相手個体から得られる利得は、同じ利得を得た個体の普及率が高い場合は低く、普及率が低い場合には高くなる。本研究では、個体 a の資源共有法による適応度 $F_{rs}(a)$ を以下のように定義した。

$$F_{rs}(a) = \sum_{b \in P_B} \frac{E(a, b)}{\sum_{a' \in P_A} count(a, a', b)} \quad (3)$$

$$count(a, a', b) = \begin{cases} 1 & \text{if } E(a, b) = E(a', b) \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

ここで、 P_A と P_B は各集団の個体の集合とする。関数 $count$ は、個体 b による評価値が個体 a と同じ個体が集団 P_A 中にどれだけ存在するかを表す。個体 b によるその評価値は、その数に反比例した値になる。

3. 提案手法

3.1 B-FS 法

フォーカシングを防止するためには集団中の多様性が重要である。そして、相手集団に適切な進化圧を与えるには遺伝子型/表現型空間における多様性よりも、個体が相手集団に対して与える特徴に関する評価基準における多様性が必要だと考えられる。

本研究で提案する B-FS (Behavior-based Fitness Sharing) 法とは、遺伝子型/表現型空間上での個体間の近さを用いるのではなく、個体の振る舞いの類似度を用いた適応度共有法である。

本研究で扱う競合型共進化アルゴリズムでは、個体の振る舞いとして相手集団による評価結果を用いる。具体的には、集団 B による集団 A の個体 a の評価結果を以下のようにベクトル t_a として表す。

$$t_a = (E(a, b_1), E(a, b_2), \dots, E(a, b_{N_B})) \quad (5)$$

ここで $E(a, b)$ は、個体 b による個体 a の評価を表す。この評価結果のベクトル間の距離をその個体間の類似度とし、適応度共有法を用いる。本稿ではベクトルの距離関数として、ベクトルの差の1次ノルムを用いた。

B-FS 法は各個体の評価結果に基づいており、そこから相手集団に対して与える評価基準の類似度を判別できると考えられる。

n 次元ベクトル \vec{p} の1次ノルム $\|\vec{p}\|_1 = \sum_{i=1}^n |p_i|$

3.2 エリート抑制

前述のように、共進化アルゴリズムにおいて、その個体が相手集団に対してどの特徴に関する評価基準を与えているかが重要である。また、優れた特徴を複数持つ個体よりも、単一の優れた特徴を持つ個体が多数存在する集団の方が相手集団に対する評価関数として適切であると考えられる。

共進化アルゴリズムにおいて単一の優れた特徴を持つ個体とは、相手集団の全ての個体に対して優位な個体ではなく、ある一部の個体に対して優位な個体であると考えられる。この相手集団に対して全体的に優位である個体を抑制するために、個体が得た正の利得 ($E(p, q) > 0$) の合計に基づいたスケールをさらに加えた。具体的には、個体 i が持つ正の利得の合計を $\text{point}(i)$ とし、個体 i の適応度 $F(i)$ とした時、その個体の最終的な適応度 $F'(i)$ は以下のとおりである。

$$F'(i) = \text{weight}(\text{point}(i)) \times F(i) \quad (6)$$

$$\text{weight}(p) = 1 - \left(\frac{p}{\text{MAX_POINT}} \right)^\beta \quad (7)$$

ここで MAX_POINT は、正の利得の合計の最大値である。本研究では、1 個体あたりの評価に相手集団の全個体を用いて評価するため、MAX_POINT の値は集団中の個体数となる。パラメータ β により、エリート個体とみなす得点の範囲を制御する。 β が高いとき、エリート個体とみなされる得点もまた高くなる。

エリート抑制は、B-FS を含めた各多様性維持の手法と組み合わせることができる。

4. ナンバース・ゲームによる評価実験

4.1 ナンバース・ゲーム

本研究では、2 人対称ゼロ和ゲームであるナンバース・ゲーム²⁾ を用いる。ナンバース・ゲームの戦略は、 n 次元ベクトルである。ベクトルの各要素は、区間 $[0, k]$ の値をとる。純粋戦略 p が q と対戦した際の利得 $E(p, q)$ は以下の式で与えられる。

$$E(p, q) = \text{sign} \left(\sum_{i=1}^n g_i \right) \quad (8a)$$

$$g_i = \begin{cases} p_i - q_i & \text{if } h_i = \min(h) \\ 0 & \text{otherwise} \end{cases} \quad (8b)$$

$$h_i = |p_i - q_i| \quad (8c)$$

ここで、関数 sign は、入力が正の数ならば 1 を、負ならば -1 を、0 ならば 0 を返す関数である。

参考文献 2) ではパラメータ ϵ が存在するが、本稿ではそれを 0 と設定している。

2 次元のナンバース・ゲームの具体的な動作の例を、 $p = (3, 6)$, $q = (1, 5)$, $r = (4, 4)$ の 3 つの関係を用いて示す。 p と q では、第 2 要素間の差が第 1 要素間の差よりも小さいので、第 2 要素の大小を比較する。よって p の第 2 要素が q の第 2 要素よりも大きいので p が利得 $E(p, q) = 1$ を得る。同様に q は r に対し第 2 要素の比較で大きいので $E(q, r) = 1$ となる。しかし、 r は p に対し第 1 要素の比較で大きいので、 $E(r, q) = 1$ となり、この 3 つの中でどれが最も優れた戦略かを判断することができないという問題が起こる。以上のように、ナンバース・ゲームは推移律が成立しない。

ナンバース・ゲームを用いる利点は、シンプルであり、戦略の絶対的な良さがわかっている一方で、推移律が成立しないことである。そのため、共進化アルゴリズムの手法を評価に適している。

本研究では、個体の世代によらない評価値である絶対適応度をその表現型であるベクトルの全要素の和とし、相対適応度を対戦によって得られた利得から計算される適応度とする。

4.2 実験設定

文献 2) に従い、以下のとおり設定した。 n 次元のナンバース・ゲームの戦略を表す遺伝子として長さ nk のビット列 $\langle b_1, b_2, \dots, b_{nk} \rangle$ を用意し、 $\langle \sum_{i=1}^k b_i, \sum_{i=k+1}^{2k} b_i, \dots, \sum_{i=k(n-1)+1}^{nk} b_i \rangle$ として表現型であるベクトルを生成する。全ビット 0 の個体を初期集団に設定した。遺伝的操作には突然変異のみとし、選択にはルーレット選択を用いて次世代の集団を生成する。また、個体の持つ遺伝子のビット列が全て 0 の状態として初期集団を生成する。共進化アルゴリズムのパラメータとして、各集団の個体数は 100、突然変異確率は 0.005、世代数は 1000 を用いた。また、ナンバース・ゲームのパラメータとして、次元数 n は 2、各要素の区間 k は 100 と設定した。

本研究では適応度共有法の距離関数として、遺伝子型にはハミング距離を、表現型にはユークリッド距離を用い、そのパラメータ σ_{share} に、G-FS では 1, 2, ..., 10, 20, ..., 200, P-FS では 1, 2, ..., 10, 20, ..., 140, B-FS では 1, 2, ..., 10, 20, ..., 200 を用いた。各適応度共有法のシェアリング関数による影響の度合いを決定するパラメータ α は、1 を用いた。また、エリート抑制のパラメータ β には、1, 2, ..., 10, 20, ..., 50 を用いた。

以上の設定で、一方の集団に各手法を適用し、25 試行ずつ実験を行った。各手法を適用した集団を集団 A、もう一方を集団 B とする。

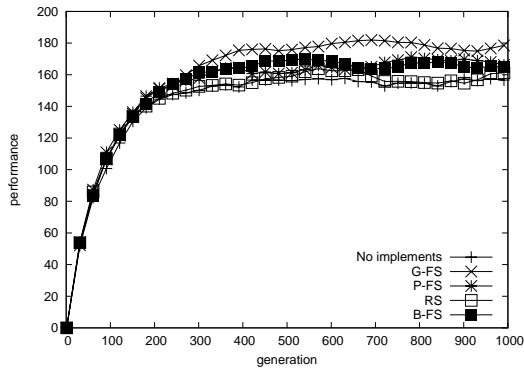


図 3 各手法における集団 B の絶対適応度の変化

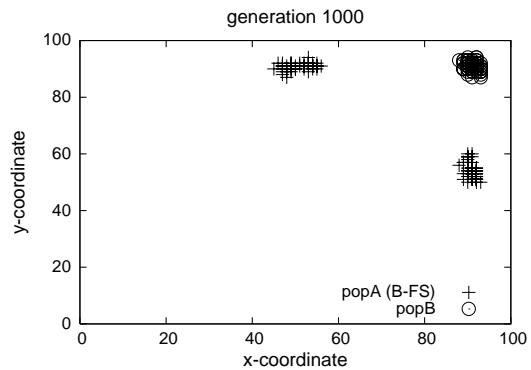


図 4 1000 世代目における個体の分布図 (B-FS)

4.3 比較評価

集団 B 中の絶対適応度の最大値の 800 世代から 1000 世代までの平均値を各試行の評価値とし、各手法のパラメータ σ_{share} にはその評価値の平均が最大であったもの (G-FS では 10, P-FS では 6, B-FS では 8) を比較に用いた。

図 3 に、各世代における集団 B 中の絶対適応度の最大値の平均を示す。200 世代以降において、多様性維持手法を適用しない場合ではフォーカシングが起きているため、160 を超えることはなかった。200 世代以降において、各多様性維持手法を適用した場合は適用しない場合に比べ、高い値を示している。

図 4 に、B-FS を適用した場合の 1000 世代目における個体の分布図を示す。この図のように各多様性維持手法を適用した場合では、集団 A 中の個体が x 軸、y 軸方向へ分かれているため、集団 B に対して両方向への進化圧がかかり、フォーカシングを防ぐことができている。B-FS では集団 A がより最適解へ近いほうへ進化することがあるため、P-FS より良い性能であったが、G-FS を超えることができなかった。

図 5 に、エリート抑制のみを適用した場合におけるパラメータ β による評価値の 25 試行平均の変化を示す。 β の値が 10 を超えると、その評価値は大きく下がる。これは、関数 weight によりエリートとみなされる個体の範囲が狭くなり、エリート抑制の効果が小さくなるためである。エリート抑制のみを適用した場合においても、図 4 のように集団 A 中の個体が各方向へ分裂する現象が確認された。これは、x と y の値が共に大きな値を持つ個体は得点が高くなりやすく、エリート抑制によって淘汰されやすくなるためである。

図 6 に、エリート抑制と各多様性維持手法を組み合わせた場合における集団 B の絶対適応度の変化を示す。各組み合わせにおけるパラメータ β は、多様性

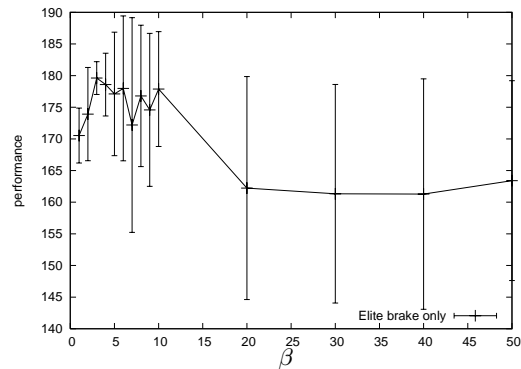


図 5 エリート抑制のパラメータ β による評価値の変化

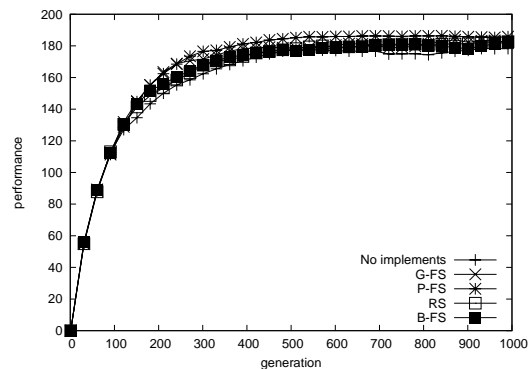


図 6 エリート抑制を組み合わせた場合における集団 B の絶対適応度の変化

維持手法を適用しない場合では 3.0, G-FS では 9.0, P-FS では 7.0, RS では 9.0, B-FS では 5.0 を用いた。先ほどの図 3 と比較すると、各手法においてその値が 180 付近で安定している。また、B-FS にエリート抑制を加えることにより、ほぼトップの性能をあげた。これは、先述の集団 A がより最適解へ近い方へ進化することが、エリート抑制により解決されたため

表 1 ナンバース・ゲームにおける各手法の評価値

多様性維持手法	エリート抑制	
	なし	あり
なし	1.577×10^2	1.796×10^2
G-FS	1.812×10^2	1.855×10^2
P-FS	1.678×10^2	1.855×10^2
RS	1.559×10^2	1.823×10^2
B-FS	1.727×10^2	1.846×10^2

である。

各手法の評価値の 25 試行平均を表 1 に示す。多様性維持手法を適用しない場合の評価値との性能比は、G-FS では 14.9%、P-FS では 6.4%、RS では -1.1%、B-FS では 9.5% であった。RS を除き、多様性維持手法を適用した場合は、その性能を向上させることができた。さらにエリート抑制を加えた場合、各手法の性能が向上した。エリート抑制を加える前後の性能比は、G-FS では 2.3%、P-FS では 10.5%、RS では 16.9%、B-FS では 6.9% となり、B-FS は G-FS、P-FS とほぼ並ぶ性能となった。

5. CA 密度分類問題による評価実験

5.1 CA 密度分類問題

セルオートマトン (CA) 密度分類問題とは、初期 (ステップ $t = 0$) において全セル中に含まれる状態 1 であるセルの割合 ρ_0 が閾値 ρ_c を超えているかいないかを判定する CA の状態遷移ルールを発見する問題である。もし超えていた場合には全セルの状態を状態 1 へ、超えていない場合には状態 0 へ M ステップ以内に CA は遷移させる。本研究では 1 次元 CA 密度分類問題を対象とし、その閾値 ρ_c は $1/2$ とする。

密度分類問題に共進化アルゴリズムを適用した場合にフォーカシングが起きることが確認されており、初期状態の集団には多様性が必要であることが指摘されている¹¹⁾⁷⁾⁸⁾。

5.2 実験設定

$\rho_c = 1/2$ の密度分類問題を、CA の状態遷移ルールと全セルの初期状態の集団との共進化アルゴリズムにより行う。CA の設定として、近傍の半径に $r = 3$ 、セルの格子数に $N = 99$ 、最大ステップ数に $M = 200$ を用いた。

文献 7) に従い、以下のとおり設定した。状態遷移ルールは近傍の状態の各パターンに対する出力を列挙することで記述できるので、その染色体を長さ 128 の 2 進数列として表す。初期状態の染色体も状態遷移ルールの場合と同様に、長さ N の 2 進数列として表現する。

状態遷移ルールの適応度は、正確に処理することができた初期状態の数である。それに対し初期状態の適応度は、その初期状態を正確に処理することができなかった状態遷移ルールの数である。

状態遷移ルールの初期集団中の個体は、区間 $[0, 1]$ の一様乱数により決定した λ パラメータに基づいて生成される。初期状態も同様に、区間 $[0, 1]$ の一様乱数により決定した密度 ρ_0 に基づいて生成される。各集団中の個体数は 100 個体とし、エリート選択と交叉、突然変異を用いて次世代の集団を生成する。エリート数は 90 個体とし、その中から適応度に比例した確率で親を選び、交叉と突然変異を用いて残り 10 個体を生成する。1 ビットあたりの突然変異確率は 0.002 とする。

以上の設定を用い、各手法を初期状態の集団に適用し、5 試行ずつ実験を行った。初期状態の個体は遺伝子型と表現型が同一であるため、従来の適応度共有法は G-FS のみ適用し、距離関数にはハミング距離を用いた。各適応度共有法のパラメータ σ_{share} に、G-FS では 5, 10, 20, 50, 99、B-FS では 5, 10, 20, 50, 100 を用い、シェアリング関数のパラメータ α には 1 を用いた。また、エリート抑制のパラメータ β には 2, 5, 10, 50 を用いた。

5.3 比較評価

各世代において最大適応度を持つ状態遷移ルールに対して、ランダムに生成した 500 個の初期状態のうち正確に処理した割合 (正答率) を計算し、それを各手法の比較評価に用いた。正答率の全世代の平均を各試行の評価値とし、各手法のパラメータ σ_{share} にはその評価値の平均が最大であったもの (G-FS では 50、B-FS では 20) を用いた。

図 7 に、最大適応度を持つ状態遷移ルールの正答率の試行平均の変化を示す。多様性維持手法を適用しない場合はフォーカシングが起きるため、正答率は 0.5 付近を保っている。G-FS を適用した場合は、正答率の上昇がおよそ 300 世代以降において見られる。それに対し、B-FS を適用した場合はおよそ 100 世代から 200 世代にかけて正答率が 0.8 以上にまで上昇する。そして、G-FS に比べて B-FS は高い正答率の状態遷移ルールを安定して得ることができた。これは初期状態の密度 ρ_0 が $1/2$ 付近に収束した際に、G-FS では $1/2$ より上または下のどちらか一方へ収束するのに対し、B-FS ではその両方に分布させることができたためだと考えられる。RS を適用した場合においても、上述の B-FS と同様の初期状態の分布が見られたが、その進化の速度が遅く、高い正答率を得ることが

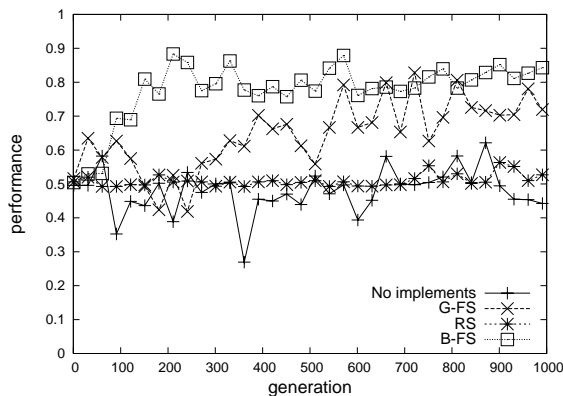


図7 各手法ごとの状態遷移ルールの正答率の変化

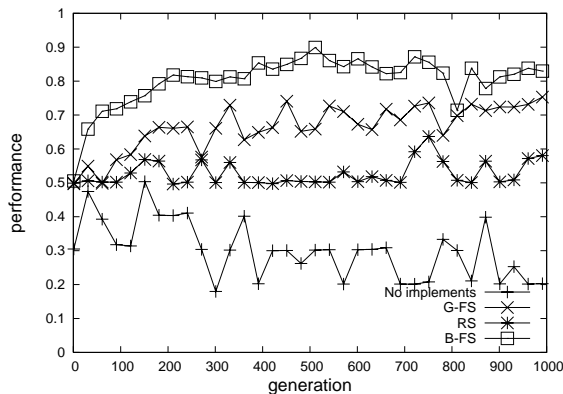


図8 エリート抑制を組み合わせた場合の状態遷移ルールの正答率の変化

できなかった。これは文献7)8)のように、RSは両集団に対し適用することでその効果が現れるためだと考えられる。

エリート抑制のみを適用した場合、パラメータ $\beta = 2.0, 5.0, 10.0, 50.0$ における評価値の5試行平均はそれぞれ0.3580, 0.3433, 0.3561, 0.2971となり、各パラメータ値において、フォーカシングが起きることが確認された。エリート抑制は相対的に難しいものほど適応度を低くするため、初期世代において相対的に難しい初期状態の集団は相対的に簡単なものへ進化する。そのため、初期状態の集団の中で次世代に残りやすい個体は、密度 ρ_0 が $1/2$ に近いものではなく、1または0に近いものである。状態遷移ルールの集団においてもまた、 λ パラメータが1または0に近いものが残りやすくなるため、初期集団中の個体数の偏りにより次世代の集団に偏りが生まれる。

図8に、エリート抑制と各多様性維持手法とを組み合わせた場合における正答率の変化を示す。各組み合わせにおけるパラメータ β は、多様性維持手法を適

表2 密度分類問題における各手法の評価値

多様性維持手法	エリート抑制	
	なし	あり
なし	0.4890	0.3580
G-FS	0.6385	0.6644
RS	0.5135	0.5268
B-FS	0.7774	0.8049

用しない場合は10、G-FSでは2、RSでは5、B-FSでは50において正答率の平均が最大となった。エリート抑制のみでは正答率が0.5より低い値へ進んでいるが、多様性維持手法を組み合わせることによってフォーカシングを防ぐことができ、高い正答率を得られた。特にG-FSとB-FSではエリート抑制を加えることにより、状態遷移ルールは早い世代で高い正答率へ進化することができた。RSでは進化の速度が遅いため、エリート抑制と組み合わせても正答率が大きく向上しなかった。

各手法の評価値の5試行平均を表2に示す。多様性維持手法を適用しない場合との性能比は、G-FSでは30.6%、RSでは5.0%、B-FSでは58.9%であった。エリート抑制のみを適用した場合の性能比は-26.8%であったのに対し、多様性維持手法と組み合わせた場合の組み合わせる前との性能比は、G-FSでは4.1%、RSでは2.6%、B-FSでは3.5%であった。B-FSとエリート抑制を組み合わせたものが最も評価値が高く、何も適用しないものに比べて64.6%の性能の向上が見られた。

6. おわりに

競合型共進化アルゴリズムにおいて、多様性を維持する機能を一方の集団に適用することによって、フォーカシングを解決しうることがナンバース・ゲーム、CA密度分類問題へ適用することによって示した。その多様性維持手法として適応度共有法を用いること、そしてその多様性の尺度として遺伝子型/表現型ではなく個体の相手集団に対する振る舞いの履歴を用いるB-FS法が有効であることを評価実験において示した。

ナンバース・ゲームにおいて、B-FSは何も適用しない場合との性能比が9.5%であり、他の適応度共有法に比べ若干下回る結果だが、さらにエリート抑制を加えることによりその性能比は17.1%まで向上し、ほぼトップの性能をあげた。一次元CAにおける密度分類問題では、B-FSのみ適用した場合と何も適用しない場合との性能比は58.9%であり、エリート抑制と組み合わせずとも今回比較した多様性維持手法の中で

1 番良い結果を得た。これは、ナンバース・ゲームでは個体である戦略の特徴は 2 次元であるのに対し、密度分類問題の初期状態の特徴である密度は 1 次元であるためだと考えられる。

また、多様性維持手法と共にエリート抑制を適用し、相手集団に対して全体的に優位な個体を抑えることで、相手集団によりよい進化圧を与えることがわかった。このことから、エリート抑制には逆噴射機能¹⁶⁾に似た効果があり、一次元 CA における密度分類問題のように、初期世代において集団間の適応度差が大きく開いている状況において、その効果が有効に働くと考えられる。

本研究で提案する B-FS 法とエリート抑制は、遺伝子型または表現型の距離関数を定義することが困難または不可能な場合においても、個体間の競合関係さえ定義できれば実装することができ、有効であると考えられる。

参 考 文 献

- 1) Barbosa, H. J.C.: A Coevolutionary Genetic Algorithm for a Game Approach to Structural Optimization, *Proceedings of the Seventh International Conference on Genetic Algorithms* (1997).
- 2) Ficici, S. G. and Pollack, J. B.: A Game-Theoretic Memory Mechanism for Coevolution, *Proceedings of the 2003 Genetic and Evolutionary Computation Conference* (2003).
- 3) Goldberg, D.E. and Richardson, J.: Genetic Algorithms with Sharing for Multimodal Function Optimization, *Proceedings of the Second International Conference on Genetic Algorithms* (1987).
- 4) Hillis, B.: Co-Evolving Parasites Improve Simulated Evolution as an Optimization Procedure, *Artificial Life II* (1992).
- 5) Jensen, M.T.: A New Look at Solving Minimax Problems with Coevolution, *Proceedings of 4th Metaheuristics International Conference* (2001).
- 6) Juillé, H. and Pollack, J.B.: Co-evolving Interwined Spirals, *Proceedings of the Fifth Annual Conference on Evolutionary Programming*, pp. 461–468 (1996).
- 7) Juillé, H. and Pollack, J. B.: Cevolutionary Learning: a Case Study, *Proceedings of the Fifteenth International Conference on Machine Learning* (1998).
- 8) Juillé, H. and Pollack, J.B.: Coevolving the Ideal Trainer: Application to the Discovery of Cellular Automata Rules, *Proceedings of the Third Annual Genetic Programming Conference* (1998).
- 9) Noble, J. and Watson, R.A.: Pareto Coevolution: Using Performance against Coevolved Opponents in a Game as Dimensions for Pareto Selection, *Proceedings of the Genetic and Evolutionary Computation Conference* (2001).
- 10) Paredis, J.: Coevolutionary Computation, *Artificial Life*, Vol.2, No.4 (1995).
- 11) Paredis, J.: Coevolving Cellular Automata: Be Aware of the Red Queen!, *Proceedings of the 7th Int. Conference on Genetic Algorithms* (1997).
- 12) Pollack, J.B. and Blair, A.D.: Co-Evolution in the Successful Learning of Backgammon Strategy, *Machine Learning*, Vol.32, No.1 (1998).
- 13) Rosin, C.D. and Below, R.K.: Methods for Competitive Co-evolution: Finding Opponents Worth Beating, *Proceedings of the Sixth International Conference on Genetic Algorithms* (1995).
- 14) Smith, R. E. and Gray, B.: Co-Adaptive Genetic Algorithms: An Example in Othello Strategy, *Proceedings of The Florida Artificial Intelligence Research Symposium* (1994).
- 15) Watson, R.A. and Pollack, J.B.: Coevolutionary Dynamics in a Minimal Substrate, *Proceedings of the Genetic and Evolutionary Computation Conference* (2001).
- 16) 三井富雄, 有田隆也: 進化圧を自動調整するホスト/パラサイト共進化型遺伝的アルゴリズム, 人工知能学会全国大会論文集, 2C3-04 (2001).